

**Atty Docket No.: IDF 1764  
(4000-06700)**

**Patent**

## **REMARKS**

Claims 1-15 are currently pending in this Application. By the office action of January 3, 2006, the Examiner has finally rejected Claims 1-15 on various grounds discussed below. The Applicants respectfully traverse these rejections. Reconsideration is requested.

### **Interview Summary**

The Applicants thank the Examiner for holding a telephone interview with Applicants' attorney Albert Metrailler on February 22, 2006. The Applicants' attorney argued that the limitation of claim 1: "verifying proper operation of the binary file" is not taught or suggested by the Rasmussen US Patent 6,640,334. In particular, the attorney argued that Rasmussen teaches only checksum testing of software to determine if it was downloaded without errors. However, Rasmussen does not teach operating software that was downloaded without error to see whether it can properly operate the system. The Examiner indicated that he understood the argument and saw the distinction. No agreement on allowability was reached. The Examiner requested that the detailed arguments be provided in a response after final and agreed to review the arguments with his supervisory examiner. The Examiner also indicated that further search may be required, but that the finality of the rejection should be withdrawn. The detailed arguments are provided below.

**Atty Docket No.: IDF 1764  
(4000-06700)**

**Patent**

### **Claim Rejections - 35 U.S.C. §103**

Claims 1-3, 7, 8 and 12-15 were rejected under 35 U.S.C. §103(a) as being unpatentable over Rasmussen, US Patent No. 6,640,334 in view of Synnestvedt, et al. US Patent No. 6,598,057 on the same grounds as were provided in the previous action of August 2, 2005.

The Rasmussen reference does not teach or suggest "verifying proper operation of the binary file". This argument requires showing a negative, i.e. that something is not present in the reference. To do this requires a review and explanation of essentially the entire reference to determine what it does teach, and thereby to show what it does not teach. The following explanation is therefore somewhat long, but is needed to support the argument.

In Rasmussen Fig. 3, the structure of the flash memory is shown. It includes four partitions or pages 28, 30, 32, and 34. An original software load 20a is placed in page 28 at the factory and is never changed. It includes boot logic 22a that is always used to boot, reboot or reset the system. Pages 30 and 32 are left empty at the factory. After two or more downloads of new or updated application logic 24b and 24c, the two pages 30 and 32 will contain the most recent download and the next most recent download. Assuming that the downloads were successful, the most recent download will be designated as the active page by the active page flag 36 which is the only data stored in page 34.

Rasmussen Fig. 4 illustrates a reset process that assumes that the flash memory of Fig. 3 has files stored in all pages, i.e. that at least two software downloads have occurred. Reset begins by starting the boot logic 22a at step 401. At 402 the boot logic

**Atty Docket No.: IDF 1764  
(4000-06700)**

**Patent**

retrieves the active page flag 36 from page 34. The boot logic then retrieves the checksum 38 from the active page and determines if it is okay. Being okay indicates that the software was downloaded without error, not that it will operate properly. If the checksum is okay, then at 404 the designated application software is started and operates the system. Rasmussen does not teach what to do if the application started at step 404 does not actually operate properly. If it does not operate properly, the system would be inoperative. The user may be able to cause another reset by unplugging power to the system and then plugging it back in, but the reset process would still end up at step 404 with an inoperative system.

At step 403, if the checksum is not okay, the boot logic tries to start the application in the inactive page. At 406 it again checks the checksum at the inactive page, but again this only indicates that the software was downloaded without error. If it is okay, the system resets the page flag to make the inactive page the active page and runs the system with what was the inactive page. In normal operation, the software that was on the inactive page should operate properly, since it had been running the system for some time prior to the latest download. But if it does not operate properly, then the system is again locked up at 407 with inoperative software.

If at step 406, the checksum was not okay, the system starts the system with the original factory installed application 24a in page 28. The original software should operate the system, but only for the purpose of downloading a proper current application software.

Rasmussen therefore provides a reset process that should handle the scenario where the downloading process failed to properly download the software as indicated by

**Atty Docket No.: IDF 1764  
(4000-06700)**

**Patent**

a checksum error. However, if software was properly downloaded, but for some reason the software will not operate the system, the system will lock up and has no way to recover. It will have to be sent back to the factory or a technician will have to manually load new software, e.g. by replacing the memory card. Generally speaking, this is the problem identified in paragraph 0046 of the present application.

The main teaching of Rasmussen has to do with the downloading process itself. Rasmussen teaches that use of flash memory is desirable. However, Rasmussen also teaches that it is not possible to read from and write to flash memory at the same time. The application operating the system handles the downloading of new software including writing it into the inactive page. But in Rasmussen the application normally operates from the active page of the same flash memory. The application cannot be operating, which requires reading from the active page, while also writing into the inactive page. Rasmussen teaches using RAM to store at least part of the application for operation while it is writing a new download into the inactive page. While operating from RAM, the system can write into the inactive page.

Rasmussen at column 10 describes the downloading process in detail with reference to Fig. 6. At step 602 the updated firmware load is temporarily sent to buffers and a checksum is performed. The checksum only indicates whether the software was downloaded without error. If the checksum was good, then at step 604 the new firmware load is copied, i.e. written, into the inactive page. At step 605, the active page flag is changed to indicate that the inactive page becomes the active page. However, the application used to perform the downloading is still operating from RAM. The system must be reset or rebooted as described above with reference to Figs. 3 and 4.

**Atty Docket No.: IDF 1764  
(4000-06700)**

***Patent***

Upon resetting, and assuming the checksum for the latest download is good, it will become the current operating application. If it does not operate properly, the system locks up.

The Examiner cited several specific sections of Rasmussen. For example Col. 6, lines 20-34, was cited for stating that: "if subsequent attempts to load updated versions of the firmware fail (and also prior to receipt of any updated versions), then the communication device 1 can operate using the original firmware load 20a stored in the Boot Page 28." While this is true, it discusses only what happens in the event of a failure to properly load new software. As discussed above, a loading failure is indicated by a checksum that is not okay. There is no teaching of operating a new download to see if it operates properly, even if it was downloaded without error. This is consistent with the above explanation of Rasmussen Figs. 3-4.

Rasmussen Col. 7, lines 26-29 was cited for stating that: "the original version boot logic 22a uses the Active Page Flag 36, the check-sum 38 and jump code 40 to access, verify and begin execution of the appropriate application logic 24." This again is true, but fails to teach operating the new software to determine if it operates properly. The boot logic uses the active page flag to identify which page 30 or 32 is now active. It uses the check sum 38 to determine if the software on the active page was downloaded properly (not to determine if it actually operates properly). It uses the jump code to find the starting address for the active application. None of these steps teaches a process of verifying that the software actually operates properly. As discussed above, failure to verify proper operation may, and has been found to actually, result in an inoperative system that cannot cure itself.

**Atty Docket No.: IDF 1764  
(4000-06700)**

**Patent**

Based on the above two citations from Rasmussen, the Examiner made several assumptions about the teachings of Rasmussen.

One assumption was that that Rasmussen attempts to operate the hub using the updated binary file. To a certain extent this is true. However, the decision as to whether to start the updated binary file is based entirely on the checksum, i.e. whether the updated binary file was properly downloaded. There is no teaching or suggestion to verify that the updated binary file can properly operate the system. As described above, Rasmussen teaches starting the updated file if the checksum is okay, but if the updated file cannot operate the system properly, there is no process for restarting with another version of the file. Instead the system will lock up.

The Examiner noted that if the updated code passes the test, then the "Active Page Flag" is set depending on the location of the valid updated binary file. While this is true, the question is what is the "test"? As discussed above, Rasmussen only teaches checking the checksum, which only tests to see if the binary file was properly downloaded. Rasmussen does not provide any test to verify that a properly downloaded file will operate properly.

The Examiner also stated that in Rasmussen, if the updated binary file fails to properly operate the hub, then the hub may revert back to the original boot code and the downloaded file is not designated as the current file for the hub. This reading of Rasmussen is not accurate for several reasons. Rasmussen only teaches use of a checksum to verify proper downloading, not proper operation. It is on reset that the system may revert back to the previous download or the original load, but it does this based on finding a checksum that is not okay, not based on failure to operate. If

**Atty Docket No.: IDF 1764  
(4000-06700)**

**Patent**

properly downloaded software is actually started in the system and it will not properly operate the system, the system has no process for reverting back to any other software. Even upon repeated resets, it would always end up with the inoperative software.

Independent claims 1 and 7 include "verifying proper operation of the binary file". Rasmussen does not teach this step. Rasmussen only checks to see if the file passes the checksum test, i.e. it was not corrupted in transmission. This does not verify that it will operate in the system.

Independent claim 12 includes the process of storing a message in volatile memory across a reboot process. Those skilled in the art would not expect this to work. However, as taught in the specification, this process does work in the present application in a customer premises hub. It not only works, but provides a fail-safe method for testing the latest binary file download before "designating the new binary file as the currently active binary file", as covered in claim 15. Neither of the applied references teaches or suggests such a process.

In view of the substantial differences identified in the above remarks, the Applicants submit that claims 1, 7, and 12 are clearly patentable over the cited references. Since the remaining claims all depend from claims 1, 7, or 12, these dependent claims should also be patentable over the cited references.

Atty Docket No.: IDF 1764  
(4000-06700)

Patent

### CONCLUSION

The Commissioner is hereby authorized to charge payment of any further fees associated with any of the foregoing papers submitted herewith, or to credit any overpayment thereof, to Deposit Account No. 21-0765, Sprint.

Applicants respectfully submit that the present application as amended is in condition for allowance. If the Examiner has any questions or comments or otherwise feels it would be helpful in expediting the application, he is encouraged to telephone the undersigned at (972) 731-2288.

Respectfully submitted,

CONLEY ROSE, P.C.

Date: 2-24-06

5700 Granite Parkway, Suite 330  
Plano, Texas 75024  
Telephone: (972) 731-2288  
Facsimile: (972) 731-2289

Albert C. Metrailler

Albert C. Metrailler  
Reg. No. 27,145

ATTORNEY FOR APPLICANT